

TP OPTIMISATION

Pour ce TP, privilège l'utilisation du terminal pour se connecter à la base de données.

Le rôle PLUSTRACE est nécessaire pour ce TP.

Vérifiez que vous avez le rôle "PLUSTRACE

```
SQL> select * from session_roles;
```

ROLE

CONNECT

RESOURCE

PLUSTRACE <= !!!

Créer les tables de travail

Script :

Exercice 1

Collectez les statistiques sur les tables EMP et DEPT. Ces statistiques permettent à l'optimisateur Oracle 'Cost Based Optimizer' de choisir le bon plan d'optimisation.

```
SQL> EXEC DBMS_STATS.GATHER_TABLE_STATS(USER,Nom_Table);
```

Vérifiez que les statistiques ont bien été collectées

```
SQL> select table_name, BLOCKS, EMPTY_BLOCKS, AVG_SPACE, CHAIN_CNT ,  
AVG_ROW_LEN, SAMPLE_SIZE, LAST_ANALYZED from user_tables;
```

Chercher la signification de chaque colonne dans la documentation.

IMPORTANT : il est important de faire la mise à jour des statistiques régulièrement notamment lorsqu'un index est créé ou lorsque la mise à jour de la table est faite.

Pour récupérer le plan d'exécution et les statistiques de consommation, exécutez ceci :

```
SQL> SET AUTOTRACE ON;
```

Exercice 2 :

- Ecrire une requête SQL pour récupérer les informations de tous les employés dont le job est 'CLERK'
- Vérifier le plan d'exécution, les statistiques et commentez.
- Générer un second plan d'exécution pour la même requête.
- Que remarquez-vous ? comment justifiez-vous ce changement ?

Exercice 3 :

Exécuter la commande suivante :

```
SQL> SELECT JOB,  
DBMS_ROWID.ROWID_RELATIVE_FNO(rowid) FICHIER,  
DBMS_ROWID.ROWID_BLOCK_NUMBER(rowid) BLOC,  
DBMS_ROWID.ROWID_ROW_NUMBER(rowid) NUMLIGNE  
FROM EMP;
```

- Quelles informations sont retournées ?
- Récupérer l'adresse physique de la ligne correspondant à l'employé 7902
- Utiliser la colonne rowid dans la clause WHERE pour sélectionner l'employé 7902.
- Vérifier le plan d'exécution, les statistiques de consommation et commentez
- Combien de blocs (db block gets + consistent gets dont physical reads) sont-ils utilisés ? Peut-on faire mieux ?

Exercice 4 :

- Créer un index sur la colonne JOB : `CREATE INDEX ix_job ON EMP(JOB) ;`
- Collectez les statistiques de l'index :
- `SQL> EXEC DBMS_STATS.GATHER_INDEX_STATS('USER2','ix');`
- Exécutez à nouveau la requête de l'exercice 2 qui vous permet de récupérer les informations de tous les employés dont le job est 'CLERK' (`SELECT * FROM EMP WHERE JOB='CLERK' ;`)
- Que remarquez-vous ? y'a-t-il une différence avec le plan le plan d'exécution/statistique de l'exercice 2 ?
- Supprimer l'index ix_job. Créer un autre index `CREATE INDEX ix_emp ON EMP(JOB,ENAME) ;` collectez les statistiques
- Modifier la requête précédente pour ne récupérer que le nom : `SELECT ENAME FROM EMP WHERE JOB='CLERK'`
- Que remarquez-vous ? comment expliquez-vous cela ?

Exercice 5 :

- Créer un index sur la colonne salaire : `SQL> CREATE INDEX ix_sal ON EMP(SAL, ENAME)`
- Exécuter : `SELECT ENAME FROM EMP WHERE sal>3000`
- Vérifier le plan d'exécution
- Exécuter : `SELECT ENAME FROM EMP WHERE sal/2>1500`
- Vérifier le plan d'exécution
- Quelle différence ? Comment expliquez-vous cela ?

Exercice 6 :

- Ecrire une requête qui fait la jointure entre les tables EMP et DEPT.
- Vérifier le plan d'exécution
- Quel algorithme de jointure est utilisé ?

Oracle utilise 3 algorithmes de tri : Nested Loop, Sort Merge Join et Hash Join. Il est possible de forcer l'utilisation d'un algorithme à l'aide de la commande HINTS.

Exécuter les commandes ci-dessous :

```
SQL> SELECT /*+USE_NL(emp dept) */ * FROM EMP NATURAL JOIN DEPT;
```

```
SQL> SELECT /*+USE_MERGE(emp dept) */ * FROM EMP NATURAL JOIN DEPT;
```

Comparer les statistiques. Quel est la meilleure option ?

- Créer un index sur la colonne DEPTNO de la table EMP : `CREATE INDEX ix_dept ON EMP (DEPTNO, ENAME) ;`
- Réexécuter la commande précédente.
- Vérifier le plan d'exécution et les statistiques.